

SyncWare News

The journal for technical
applications of T/S
computers

Volume 5 Number 4

Mar.-Apr. '88

The WINTERFEST

Hats off to the T/S User Groups of Florida, for their great job in staging the Orlando show. The production was a joint effort by four users' groups, under the spectacularly able management of Mary Lynn Johnson, with Mel Nathanson as co-pilot. (More detailed comment on succeeding pages.)

A NATIONAL USERS' GROUP

One of the more provocative ideas emerging during the WINTERFEST was that of a national Users' Group. The project is still flexible enough so that you can get your own ideas into the hopper. Think it over for a few minutes, then send your ideas to

Mary Lynn Johnson
190 Hickory Woods Court, Unit 3C
Deltona, FL 32725

or

Mel Nathanson
7515 Arbordale Dr.
Port Richey, FL 33568

Comments are wanted on all phases of the undertaking: purpose, organization, management, relationship to local users' groups, and probably many more that I haven't thought of.

This is your chance to be a part of something BIG! How the Group takes shape is important to you! Make your thoughts known.

THE RIPPLES SPREAD

Time Designs Magazine, a TS-oriented periodical that would be a fine addition to your library (AFTER SyncWare News, of course), has also been getting comments that were provoked by our reference to PC's. It's not at all surprising that their readers have just about the same spectrum of views that ours do. See the inside pages for a continuing sampling of our mail.

For Your Support

NICHOLSON NIGHTTIME NETWORK is a free BBS with a Timex sort-of SIG operated by sort-of SYSOP Fred Nachbaur. (604) 354-4666 is the number to call to get access to the latest hi-res programs, files, etc. The BBS operates 1800-0900 PST every day and all day Sunday. Set your modem for 8-N-1 and give it a try. Complete details in a forthcoming issue of SWN.

(Note: we don't know how this schedule will be affected by Daylight Savings time, which goes into effect in most US states on the first weekend of April--bw)

JOHN McMICHAEL, 1710 Palmer Dr., Laramie, WY 82070, is marketing hardware and software to interface the TS2068 to a Commodore 1520 Printer/Plotter. Prices for the hardware range from \$14.95 for a bare board to \$30.95 for an assembled and tested unit. The interface hardware includes driver/demo software. Other software packages include Pic-Plot, an auto screen dump utility, Bannerific, banner making software, and CMS/1520, a compatibility patch for Customized Mscript V5 or V5.2. Software is priced at \$8.95 each, ppd.

AFR SOFTWARE (R)

Presents:

Powerful And Inexpensive
Business Software
For "Timex-Sinclair"
Computers

WORD PROCESSING

T/S-TEXT 2000\$19.95
ZX-TEXT\$19.95

SPREADSHEET CALCULATOR

T/S-CALC 2000\$19.95
ZX-CALC\$19.95

CYCLE ACCOUNTING

T/S-ZX Financial Report Generator\$29.95
Printout Of Same\$13.00

APPOINTMENT SCHEDULER

T/S-CALENDAR 2000\$19.95
ZX-CALENDAR\$19.95

Send S.A.S.E. For Free Catalog
Or Check Or Money Order To:

A.F.R. SOFTWARE
1605 Pennsylvania Ave.
No. 204

Miami Beach, FL 33139
(305) 531-6464

"FLORIDIANS ADD SALES TAX"
Dealer Inquires Invited

LEMKE SOFTWARE, 2144 White Oak, Wichita, KS 67207, has put together the Pixel Print Press, a newsletter highlighting Lemke Desktop Publishing software programs for the TS2068. Products shown include the popular Pixel Sketch & Graphics Editor, the new Pixel Print, an Icon Package, a Font Package, a Pixel Print Utility for Tasword, and Checkbook Master. Send a SASE for the latest Lemke Software info.

GREY & CLIFFORD COMPUTER PRODUCTS, P. O. Box 2186, Inglewood, CA 90305, (213) 759-7406, announces the release of the latest version of SPECTERM-64 (TS-4.1). This latest release is available on either JLO Safe (v 2.3 or later) or AERCO disks. \$35.00 + \$3 S&H. Also new to the Grey & Clifford line of products is TMX-64, a 64 column BBS software package for the TS2068. The package is said to allow you to set up a 300/1200 baud BBS system on your TS2068 computer. Written by Kurt Casby, the price is \$30.00 + \$2 S&H. Call for details.

DAN ELLIOTT, Rt 1 Box 117, Cabool, MO 65689 is offering service for several different makes of home computers, including the Timex-Sinclair line. Prices vary, so contact Dan to get the latest info. He is also offering to make mods to TS computers and peripherals per published articles.

RUSSELL ELECTRONICS, Rd. 1 Box 539, Centre Hall, PA 16828, has published its January 1988 catalog. Our copy included the note: "We will continue to make WINKY boards, ROMSWITCH, ZXLR8 (now on EPROM), and Speech Recognition System."

KNIGHTED COMPUTERS, 10 Canalview Mall, Fulton, NY 13069, (315) 593-8219, announces its new catalog.

LARKEN ELECTRONICS, RR 2, Navan, Ontario, CANADA K4B 1H9, (613) 835-2680, offers a disk operating system that is making a strong bid to become de facto standard for the TS2068.

STEVEN PETROVIC, 1007 Joseph St., Apt. 2, Lafayette, IN 47905, is completing MAILLIST for TS1000. A label-making program listing about 150 people with 16K memory, over 350 with 32K, with search capabilities. Can do lower case if connected to a full-sized printer. Write to Steven for details.

B & G ELECTRONICS, 614 Hollywood Ave., Suffolk, VA 23434, offers a full TS2068 system, with AERCO Centronics printer interface, some software, and back-dated magazines.

FORUM

The Winterfest

Economists are always being quoted as saying that the best time to buy anything is last year. This year, they were wrong. The prices for QL's, 2068's, and a lot of peripherals, software, and literature were lower than I have ever seen them.

The weather of Sunny Florida was a bit of a disappointment, but then I guess that they need a little rain now and then to make the oranges. (Actually, the Florida residents were happy that it was raining--that part of Florida has suffered badly for lack of moisture.)

The show opened with 17 exhibitors, including SyncWare News. As of this writing, I don't have final attendance figures, but 167 visitors were clocked during the first day. As Legislator Murphy (the one who wrote Murphy's Law) would have predicted, the visitors always seemed to come in bunches, much like a platoon of traffic on the highway. But the bunching didn't keep them from buying.

The Marriott Hotel is not one of the world's lowest in price--but then, it's not one of the lowest in quality of service, either. And Disney World and EPCOT provided an interesting break for those who had time (and will power) to tear themselves away from the computer exhibits.

The customary series of lecture/seminars rounded out the event.

1500 Monitor?

Gerard Trippertree of Little Ferry, NJ wants to know:

"Does anyone have any suggestions on how to modify the TS1500 so that it will produce a composite video output to drive a Zenith monochrome monitor with a 75-ohm input impedance?" He goes on to report "great results" from his ZX81 and an AERCO composite video circuit, but finds that similar techniques will not work with the 1500.

He also would like to get a "good, clear copy" of the TS1500 schematic.

Can anyone out there help? If any of you could give us a good quality TS1500 schematic, SWN would be glad to publish it, provided that there will be no copyright complications.

A Further Word To Authors

As part of our cost-shaving program, we are going to have to reduce the payment we make to authors.

You have probably noticed that this issue carries a lot of material written by in-house people. We believe that the material is still of high quality, and will be interesting to you, the readers.

However, we don't want to be guilty of denying prospective writers the forum that we have provided

for getting their works before the public. We welcome submissions as much as we ever did, but I'm afraid that we will have to pay our authors largely in terms of immortality.

A word of encouragement to you authors: we are not going to discontinue all payment. We will continue to pay you on the basis of \$10 per page, pro-rated for segments less than a full page. And discounted, as our payments always have been, for articles that require extensive editorial work.

We hope that you budding (or fully florescent) authors will not be discouraged by our new policy. We hope that the fact that we offer a first-class medium for your manuscripts will be more important than the reduction in tangible reward.

SyncWare News

Jeffrey D. Moore	Publisher
Basil Wentworth	Editor
Fred A. Nachbaur	Technical Director
Thomas J. Bent	Assistant Editor

All subscription information, billing, and ad artwork should be addressed to Jeff at:

SyncWare News
602 South Mill Street
Louisville, OH 44641
(216) 875-1257

Article submissions, Forum, Support listings, Classified ads, and really easy questions should be addressed to Basil at:

SyncWare News
1413 Elliston Drive
Bloomington, IN 47401
(812) 336-0837

Real tough questions, matters concerning hardware, and everything else should go to Fred at:

SyncWare News
C-12, Mtn. Sta. Group Box
Nelson, BC V1L 5P1
CANADA
(604) 354-3858

Back issues of SWN are available for \$3.50 each. Vol. #1 (for the 1000) is available for \$16.95, US postage included.

SWN is done entirely on TS computers. Submissions are preferred as word processed text files recorded on good quality tape. Memotext for the 1000 and Mscript for the 2068 are the preferred word processors. Documents submitted in Tasword Two or other word processors must be done in 52-column format. If you do not have a word processor, but you feel you have a good idea, then don't hesitate to work it up and submit it. Complete writers' guidelines are available upon request from the Indiana editorial office. Advertisers' rate cards from the publisher, at the Ohio address.

SWN pays authors about \$10 per page for original articles. Payment is made at time of publication.

More on the PC

You are still commenting on the PC question, and your comments are still as diverse as ever. Several of you, however, have managed to make constructive suggestions that had never occurred to us. We're grateful for them, and we'll do what we can to carry them out. Please keep your comments coming--on this subject or on any other.

Here are excerpts from the latest:

Hal Bellison, Troy, NY--"SWN has been a good journal...if you reduce the space that points at my system (2068) you do lose me..."

Rick Cavaness, Farmington, NM--"Please get serious!...We don't need another PC mag."

Louis Dooley, Ocala, FL--"If you must branch out, make it ATARI, or APPLE, or even Commodore (AMIGA)"

James Eisfeld, Placitas, NM--"If it must go toward a 'Major', let it be the Apple II family."

W.D. English, Orange, CA--"Sorry, but I am not renewing...I've enjoyed your mag, but I'm now into 32 bit words, 2 meg memory, and powerful languages..."

But I wonder whether he'll have so much fun-bw

R. Arthur Gandin, Petosky, MI--"It's really hard to believe that you're planning to support the IBM PC...You should be looking for other uses for the ZX81 and 2068."

A constructive suggestion-bw

Bob Howard, West Covina, CA--"...don't waste space supporting PC's, take space to discuss why one would want or not want one, along with discussion of what is the best upgrade...(such as QL, AMIGA, ST)"

Jim Kopisch, Oklahoma City--"Our fate may be the 'BIG BLUE' PC, but please, don't rush it along." However, he suggests articles on hardware and software that can be modified to use with TS computers. "Also, new and better methods of interacting between different types and brands of computers."

Donald Lambert, Cedar Rapids, IA--"...the article on IBM PC just overloaded my mind...if T/S is dropped from SyncWare News, I will no longer subscribe...I'll never live long enough to exploit all the features of the T/S computers I own."

Bill Manschot, New Orleans, LA--"I'd like PC articles that are pertinent to Sinclair owners." He would like to see articles on MSDOS and other commands or procedures not used on the Sinclair computers, with hints on how to translate them into Sinclairese.

John J. Shephard III, Coldwater, MS--"Your doom and gloom obituary made me mad enough to chew nails...don't let them die, fix them." He suggests that we develop self-diagnostic software, as well as software that would let a 2068 perform diagnostic tests on a TS1000, and vice versa. About the PC: "if the articles are for education, to aid in becoming a more informed member of the computer

hobby community or help in using PC advantages in the use of our T/S computers--then the articles are sure to be welcome." But not to help change over to PC. "...only use the PC as it can help us use our T/S computers."

Now, there's a really constructive suggestion--in a letter that is uniformly upbeat in tone. Does anyone have ideas on self-diagnostic programs?-bw

Guilford Taylor, Baltimore, OH--"Lord knows there's enough PC publications out there already...If you do decide to run articles on the IBM PC, I hope they are of a comparative nature that would be of interest to Sinclair users. A perfect example might be to compare IBM BASICA to Sinclair Basic."

Another fine suggestion--any takers?-bw

Van Vangor, Island Falls, ME--"I don't give two hoots about PC's...There are few T/S publications left for one to turn to."

NOTE: Van also made the suggestion that served as the basis for the IFWARE™ proposal elsewhere in this issue.-bw

Bret Lanius, Decatur, GA--"As long as SyncWare doesn't get to be dominated by PC articles, I'll be happy! I own an IBM PC but I don't program on it much. I prefer to 'tinker' with my 2068."

Don Mattocks, Sun Lakes, AZ--"My Timex-Sinclair died...(but)...I don't want to lose contact."

Arthur B. Marzano, Philadelphia, PA--"I love my TS's and encourage all readers to submit any material which may be useful to the loyal TS fans left...keep our journal TS oriented."

Things To Come...

April 17, the second Practical Computer Show, Ross Park Mall, Pittsburgh. For more information, call Dreux Priore, 102 Arbor Drive, Pittsburgh, PA 15237, telephone (412) 364-2571.

Late May-June. We hear reports of the possibility of a show in the New Jersey/New York area. If the show's organizers will keep us posted, we'll be glad to pass information on to our readers.

August 27/28, Cleveland. For more information, call Andrew Kosiorek, (216) 226-4503.

August 6/7, the 3rd Annual International/Great Northwest TS Mini-Fair. This promises to be a skookum event, as they say out in those parts. Rod Gowen is Chairman of the event, and the mailing address is 1419-1/2 Seventh Street, Oregon City, OR, telephone (503) 655-7484.

Information available to date indicates that the Northwest Fair will be held at the Cosmopolitan Hotel in Portland, 15 minutes from the airport, 2 blocks from I-5. Hotel rooms go for \$38 (plus 9% tax) for single or double. Admission is \$3, tickets at the door.

Exhibitors' rates are \$20 for 2 tables, \$5 for each additional table--one free pass given for each table. Special rates for users' groups: \$10 for 2 tables, \$3 for each additional table.

Twelve seminars are planned, on such subjects as 68000 architecture, Z80 machine code, CP/M on TS computers, GIF graphics, telecommunications, and other equally fascinating subjects, with a roundtable discussion at 9 on Saturday evening, with seminar speakers fielding questions from the floor.

A tentative list of seminar speakers includes such well-known names as Jack Dohany, Norm Lehfeld, Ed Grey, Mike de Sosa, Syd Wyncoop, Vince Lyon, Wilf Rigger, and Michael Carver. Sponsors of the show include CCAT/S users' group, R&G Enterprises, and Time Designs Magazine.

For your family, the city of Portland offers the choice of a number of delightful activities. And if you have the time to drive around the vicinity, you will enjoy some of the most beautiful scenery in the United States.

Something On Account(s)

One of the moves triggered by our recent change of personnel was a thorough (and somewhat overdue) audit of our accounts. Our findings have prompted us to make a number of changes in our practices. We believe that none of these changes will reduce the quality of our magazine, or of our services to you, our readers. Some of them may, in fact, work to your advantage.

We find that costs have been creeping up on us, unobserved. At the same time, revenues have not grown accordingly. Clearly, this is no way to run a railroad. You've heard about the company that loses a little on every transaction, but makes it up in volume--but we haven't figured out how they do it. Not yet.

Our first decision was NOT to raise our subscription prices at this time. We're going to stick with that decision as long as we can. With life being what it is, the time will inevitably come when we will just HAVE to raise the price. You'll remember that even the United States Congress awarded itself a pay raise last year, in order to keep up with inflation.

But we hope that we can do better than Congress did. We have held the line ever since we started publishing Volume II, and we'll resist any rise as long as we can. But if you want to take out insurance against that inevitable rise, renew early, at today's prices.

If you are worried about the remote possibility that we might close our doors some day and leave you holding the proverbial bag, let me tell you a little about our operating practices. In common with many other reputable publishers, we keep in a reserve bank account all of the money that you have entrusted to us. As each issue is mailed to you, we release an appropriate amount from that reserve, since we have fulfilled the obligation represented by that amount.

The money released, of course, goes to pay the expenses of the next issue, as well as our general expenses of operating during that period.

So if (Heaven forbid!) we have to close our doors some day, the money will be there for return to those who trusted us. We won't have to pay you off in apologies, in subscriptions to magazines that you don't want, or in other schlock.

So What To Do?

Obviously, there are two ways to ease the squeeze between income and expenses. We're trying both.

EXPENSES--we're doing all we can to cut our overhead expenses to the marrow. This sort of thing should come naturally to me. Both of my grandfathers were Vermont farmers, for whom frugality was a matter of survival. And it needn't concern you, unless you can pass along some ideas. If you do have some ideas, let's have them, by all means. They may be of interest to our readers in general, too.

One change that will affect both you and us is that we are going to start using a bulk mailing system. It means a bit more work for us, as we have to sort the outgoing mail by ZIP code. Not too much more work, though--after all, we ARE a computer-based operation.

And how does it affect you? Under the old system, if a subscriber moved, the Postal Service would forward his copies of SWN for a year...and would never tell us! I don't know what would happen after that year, but I hope that the subscriber would renew his subscription in the meantime, with a notice of change of address.

With bulk mail, the P.O. declines to forward your copy...but they do let us know what new address you have filed with them. We'll then check with you, and send you a replacement copy after you have confirmed your new address. This means that your copy will be delayed, UNLESS you let us know ahead of time of your new address. Obviously, you and we would all benefit if you did give us advance notice.

What happens to your unforwarded copy? In theory, the postman throws it away. I'd like to think that he gives it to some school instead, or maybe takes it home to his kids. But we may never know the answer to that one.

RECEIPTS--If we're not going to raise the subscription price, there remain at least three other ways to increase revenues. We're going to try all three.

1. We hope to increase our subscription base. Maybe you can help us here. Tell your friends about us--show them a copy--even lend them a copy, if you can trust them to give it back. This is sort of like the pet store owner saying, "Take the puppy home for a day or two, and see whether you like him." We're confident that to see SyncWare News is to want a subscription.

So please help us to interest new readers. You'll be doing them a favor. And at the same time, you'll be helping to postpone that inevitable day when we'll have to raise the price of our subscriptions.

2. We plan to solicit more advertising. Somehow, we've let our efforts in this direction lapse. I used to love to read the ads--it reminded me of the Sears catalog at Christmas time. I'd like to increase them for the sake of making SWN more appealing to our readers. But we don't mind getting paid for the ads, too.

3. We hope to publish a few other documents that we think you would like. We propose to offer them on an I/FWARE® basis, which is described elsewhere in this issue. We'd appreciate your comments on what reprints (or what new material) you'd like to see.

We'd like your advice on anything else, too. We want SyncWare News to continue to be YOUR magazine.

TS1000 Bank Switching

Has SWN done any articles on bank switching with the TS1000?

Is it possible to make 64K NVM in 16K blocks so you would have access to four programs?

E. Ferguson
Ansley, NE

FRED REPLIES:

We've received several inquiries about bank-switching on the ZX81 family. It appears that bank-switching is an idea whose time has come.

The answer is that bank-switching IS possible, and also relatively easy. The three essential elements are: memory, decoding, and paging. The need for memory chips is obvious; recent reductions in the cost of the 64256 32K SRAM (static RAM) make non-volatile, bank-switched RAMDOS arrangements very appealing. The same holds true for the EPROM equivalent. The decoding is simply to "tell" the memory chip(s) where they should "live" in the Z80's memory map. "Paging" involves relatively simple circuitry to control which of several devices (i.e. RAM or EPROM chips) is active at any given time.

The Paging aspect has been touched upon in SWN Vol. 1. As a point in fact, this was an early version of Dr. Paul Hunter's bank-switch system, which was subsequently covered in other magazines. Though this system certainly works, it has a few drawbacks. It used 8K chips, which are no longer cost-effective for such projects. More importantly, it was I/O-mapped, meaning that you needed to install a machine-code routine after power-up in order to control the banks.

The good news is that Wilf Rigter, Harry Slot, and the other wizards at the Vancouver Sinclair User Group have developed a remarkable bank-switch board that will work with the ZX81, TS1000, TS1500, and PC8300. It uses only two support chips in addition to the eight 32K RAM or EPROM chips, for a total of 256K of non-volatile memory. It is memory-mapped, meaning that you can change banks with a simple POKE. The location is in the ROM, so it won't interfere with other peripherals.

We may be tipping our hand by reporting the existence of this, but the news is just too good to keep under my hat. The developers have given me

their assurance that as soon as the final touches have been made, the full details will be submitted to SWN for publication. There is also a strong probability that kits, bare-boards, and assembled units will be made available to those with solderphobia.

Regarding the PC8300, I can't resist the urge for a little "plug". This machine truly entered the "real world" with my development of a Sinclair-compatible ROM, which runs all known Sinclair programs except high-resolution, and sports a few other neat features. I wouldn't say that this machine will "replace" the ZX81, as much as it will "augment" it.

Most hardware for the ZX81 should work with the PC8300, if the missing control lines are brought to the edge connector. One in particular, M1, is needed for 64K RAMpacks. Incidentally, "M1* decoding" [see SWN Vol. 2:5, page 10] is NOT needed to run machine-code in 32-48K on the PC8300. -fn*

Hi Rez To Disk

I have been trying to transfer your (SMC's) SCRAM (hi-res) programs from audio cassette tape to 5-1/4" floppy disks, with the LARKEN disk system for the ZX81/TS1000. Some of the programs have a lot of BASIC lines, and these accept the LARKEN line changes and additions and work without difficulty. Some programs, such as HRCHES (SWN 4:5) which only has a few short lines of BASIC, has to be saved to disk as-is, and has to be started with a GOTO xxxx command, does not auto-run, or provide screen prompts to change the enable/disable switches, etc.

How can I make such software auto-run on the Larken disk system?

Robert Shade

FRED REPLIES:

Let's briefly review the ZX81/TS1000 memory map. At the "bottom" of the memory, at the 16K mark, are the system variables, followed by the BASIC program listing. Above that is the display file, which "floats" as the BASIC area changes in size. Beyond the display file are the BASIC variables.

There are two problems that can occur in some software. Very long programs can be "pushing" 16K (or 64K) to the limit, and adding any BASIC lines can cause trouble with "out of memory" reports. With such software, you'll have to "trim some fat" elsewhere, such as deleting un-needed REM lines. Many other memory-saving techniques have been published over the years. (See "Byte-Pynchers" in SWN Vol. 1.)

The problem with the Psion CHESS program is a different one. This program has plenty of "headroom", but employs an unusual approach to storing machine-code. Instead of being in a 1 REM or 0 REM line at the beginning of the program (fixed address), it is in a VARIABLE. This means that it moves around as you change/add/delete lines of BASIC. If the resulting address is off by even one byte, a crash is guaranteed.

With some programs that use this approach (e.g. Softsync's Asteroids), you can "trim fat" as before

to keep the program area exactly at the same length as it was before the changes. Here's a technique that might help:

Before making any changes, peek VARS by entering PRINT PEEK 16400 + 256*PEEK 16401. Write this number down. After making your changes, again peek VARS and make up the difference by changing dummy REM lines, etc. As a point of reference, a single-command line (like an "empty" REM) takes 7 bytes. Don't forget that every number in a listing takes six more bytes than show on the screen.

CHESS is a rough one, probably the roughest; there is no easy way of gaining the needed bytes. With such programs you HAVE to get creative. I'll give the following solution without proof; study of WHY each change was made, and HOW it conserves memory, might be helpful with other such programs.

After making the high-res mods detailed in SWN 4:5, the main CHESS program listing looks like this:

```
1 SAVE "CHESS"
2 LET T=USR 24692
3 INPUT A$(1 TO 9)
4 LET T=USR A
5 LET T=USR B
6 GOTO 3
```

If you peek VARS, you'll get 17393. The LARKEN-modified program looks like this:

```
1 RAND USR C
2 REM SAVE "CHS.B1"
3 PRINT Z$
4 PAUSE E
5 LET T=USR D
6 INPUT A$(VAL "1" TO VAL "9")
7 LET T=USR A AND USR B
8 GOTO F
```

After making the changes, verify that peeking VARS still gives 17393. Then enter the commands,

```
LET C=14336 (LARKEN DOS entry)
LET D=24692
LET E=40000 (to pause forever)
LET F=6
LET Z$="TURN OFF DOS, ENABLE SCRAM"
```

As before, the HRCHESS portion must have been previously loaded and safely stored in your SRAM. However, we can't use our nifty GOTO USR SAVE to bypass Psion's protection. So simply disable the SRAM, enable DOS, and GOTO 1. After saving, (and subsequent reloading), reverse the procedure then press a key. You will get a "LOAD CHECK FAILED" message, don't worry about it. The game will play just fine.-fn

Wish Lists

There is one feature that the ATARI has that I wish could somehow be programmed into the TIMEX--the attract mode timer and flag. When the computer is on but no keyboard input is read for 7 to 9 minutes, the attract mode rotates colors on the screen at low luminance levels and helps to save the TV screen from "burn-out" damage.

How about an article on, or just a list of, books in order of learning importance, from BASIC to machine language, for those of us who didn't have

someone to advise us on how to proceed without leaving gaps.

How about some serious reviews of peripherals?

Dorothea Bundy
Ventura, CA

On the subject of PC, Miss Bundy comments: "I say forget it. Now if it's a way to make the Timex compatible with others, then that's different, but I say stick to giving us what we as subscribers are paying for."

More Wishes

Hardware is the aspect of the 2068 I've always been interested in. I would be willing to buy plug-in peripherals to increase the real world usefulness of my 2068. I would like to store the city map in my computer.

Pete Kelly
Petal, MS

I particularly like articles on hardware and interfacing.

Harry T. Hall
Flagler Beach, FL

Now, there are some ideas for you serious TS programmers to sink your teeth into. Let's hear from you. -bw

Hardware BUG ALERT!

Donald Lambert, of Cedar Rapids, IA, warns that some PC8300's come with 220-volt power supplies. If yours is that way, it's possible to make a step-up auto-transformer out of a couple of filament transformers. Possible, but a nuisance. Try first to get the supplier to replace your power supply with one that's designed for North America.

While we're talking about suppliers, let me toss an unsolicited bouquet to

Sunset Electronics
2254 Taraval Street
San Francisco, CA 94116

When I bought a 2068 as a birthday present for my son, they somehow sent me a defective power supply (after all, who ever thinks to test a power supply?) When I told them, they IMMEDIATELY sent a replacement, no questions asked. They didn't even ask me to return the defective one. (And before you ask, that was before there was any hint that I was going to become editor of SWN.)

2068 REVIEW :

HI★RES

Plotting on the TS2068

Carol and Frank Davis

There was once a time when I had a small touch of envy when I read an ad for a plotter for some computer other than my Sinclairs. I was even able to use a plotter-printer with my old Commodore VIC-20. In fact, I still had my Commodore 1520, but had rarely used it the last few years. It had come with no ready software, and the type-in programs that were in the manual were riddled with errors. The plotter-printer worked fine, but I just had no real use for it.

Well, along comes

John McMichael
1710 Palmer Dr.
Laramie WY 82070

to the rescue with both an interface and software to boot.

The plotter uses plain paper that is 4.5 inches wide and comes in rolls. It is readily available at Radio Shack stores who sold a similar plotter printer a few years ago, along with Atari. They also sell the four colored pens used by the plotter. The plottable area on this paper is as follows:

"Along the X axis: 96 mm (480 positions numbered 0 to 479 separated by 479 steps)

"There is a 9 mm border on the left and right sides of the X plottable area so that the actual paper width is 114 mm.

"Along the Y axis: Programmable as desired (+/- 999 positions numbered 0 to 998 separated by 998 steps)"

With the interface of John's you are able to use this four color (red, black, blue, and green) plotter with your TS2068. He has a simple interface which will allow you to connect it to the computer in one of two ways, so let him know which you want. It can be connected directly out the back of your expansion port on the rear, or plugged into a John Oliger Co. motherboard. As I use the Oliger Disk system, I opted for the motherboard plug in. The interface comes in two forms. One is a kit, and if you are handy with a soldering iron and have the time it will save you a few bucks. Otherwise you had best order the completed board.

So what are the drawbacks to using this Commodore plotter-printer? The first is that like our Timex-Sinclairs...the 1520 is also an orphan in a sense. The company that made them is still around but does not seem to offer any support. They are available at many TOYS-R-US stores for about \$30. I have also seen used ones for sale at EVERY Ham and Computer show I went to last year. You might also check any local Commodore User Group in your area.

They are still being sold in Britain; I have seen ads for them in British computer mags such as Your Computer.

So you need to get one of the Commodore 1520's to use the interface and the software from John McMichael. I could also wish for a plotter that uses regular size paper, but this is the best deal I have run across yet.

Now to tell you a little about the software available for this plotter. First there is the Driver Program that comes with the interface, and contains a very extensive showing of just what can be done with the plotter. The demo lasts over 13 minutes and shows you geometric designs; 20, 40, 60, 80 columns for text; rotated text; different colors; and many sizes of print to choose from.

He also has available a Screen-to-Plotter program that allows you to use the plotter like a Drawing (with text) program by either using the cursor or a joystick. You can save screens, load in previously saved screens, edit them, all in four colors. When using text, you can change size of print, color, rotate print, use solid or dashed lines, center text, etc. All of this for \$8.95.

Bannerific is a banner program for the 1520. You may print out your font and size selected message either as horizontal or vertical print. Each letter can be a different color from its neighbor letter. You may also do an inverse of the selected characters. The one thing I wish it did have, but does not, is a selection of fill patterns for the text. It will not accept "token" characters, nor will you be allowed to exceed 200 characters in your text. These drawbacks are minor compared with what you get, again for only \$8.95.

The last of the currently available programs by John for the 1520 is CMS/1520. This allows you to combine MSCRIPT Version 5.0 or 5.2 with the picture-plotter as an 80 column word processor. John gives you full instructions on how to combine these two programs and to get you familiar with the ensuing differences. Some of the differences outlined are that two home menu commands are no longer effective, but are no longer needed. These are LFEED and INTERFACE, and are overwritten by the CMS/1520 driver code. This is the most fun I have ever gotten out of MSCRIPT (of course, I usually use my personalized Spectral Writer as my word processor). This program also retails for \$8.95.

I have tried to give you the honest pros and cons of this interface and software so that you can make an intelligent decision. Personally, I feel that it is a bargain and will open up a new area of computing for you.



2 7 3 9 5 4 1 0 6 RANDOM NUMBERS 8

Frederick M. Lewis

A TS2068 contest entry

This program is of truly heroic dimensions. Since it is unlikely that any of our readers would want to key in the 35,665 bytes of the program, Mr. Lewis has agreed to supply tapes of the program for a charge of \$8, which includes postage and handling within the US and Canada.

Send orders to:

Frederick M. Lewis
5 Sherwood Park Drive
Burnt Hills, NY 12027

The program includes a wealth of descriptive material and illustrative examples, as well as a useful bibliography.

The program is best described in Mr. Lewis's own words:

This program is an interactive tutorial on random numbers, ways to generate and test them and on uses. Random numbers are not at all as simple as they seem intuitively to be. They are fascinating and allow us to solve complex mathematical and physical problems that could not otherwise be treated. Although we tend to think casually about them random numbers are difficult to define, impossible to test, impractical and difficult to generate by physical means (eg 'true' random numbers as by radio-active decay) and impossible to generate by computer except as so-called pseudorandom numbers, which are not really random but hopefully 'act like' they are.

We cannot really 'test' for randomness because a true infinite series will contain all possible sequences (called global random sample) but we must use a finite sample (called a local sample). If a coin flipped heads 1000 (or one million!) times we might be suspicious but could not rule the coin non-random. What we do is test a finite sample by selected tests to see if the numbers behave locally. Certain number sequences such as PI have passed all possible tests but cannot be called random because the next digit is always predictable.

Many pseudorandom generators have been devised for computer use. All have deficiencies such

as short 'periods' after which they repeat.

Selected demonstrations from the program are described below.

HEADS OR TAILS

This program converts random numbers to 1's ($n < 0.5$) or 0's ($n \geq 0.5$) to analyze a string of random numbers either in the computer generated form (0.nnnnn) or as single random digits keyed in individually at the user option.

Another program...

... plots a field of random x/y points to test for 'order' as revealed by any systematic pattern. This method shows order not detected by other tests. The patterns developed for single generators (test using plot 1 program) show that no generator is perfect and all develop some order. This program uses two generators to compute alternate numbers switching from the built in random generator to a program generator.

One routine plots Brownian trails, while another discusses tests for randomness:

One method used to test random-number samples is based on comparing the observed frequencies of occurrence of selected digits (one at a time) with expected frequencies for randomly distributed samples. The "chi square" distribution provides a test.

The need for this is illustrated by the fact that apparently random samples are often "skewed". Consult any table of statistical data such as a list of populations of cities (any almanac) and note that the first digits are predominately digits 1-4 vs the less frequent digits 6-9.

One of the most spectacular demonstrations in the program, which generates Figure 1, is described as follows:

Process optimization is an important application. The demo simulates an actual optimization model that was designed to improve the performance of a polyurethane foam surfactant which is a function of the size distribution of foam cells. The 3-d surface represents the 'quality' as the dependent variable of a

multiple regression analysis of quality vs reaction parameters. Optimization is achieved by estimating the position of the hill top performance via successive approximations to make the best product. The actual analysis was done on a main frame and lengthy for demo. So, though a 2068 can do the job it is only simulated in this program.

One of the most impressive parts of the program is the well-researched and extensive bibliography:

The Art of Computer Programming, Ed.2, Donald E. Knuth, Addison-Wesley, Vol. 2/Seminumerical Algorithms. (One of the best).

An Introduction to Probability Theory and its Applications, William Feller, Vol. 1, Wiley, 1950, 1957, 1968 (this book is a timeless classic, it is still republished after 35 years- they call it simply "Volume One").

Computer Simulation Techniques, Thomas Naylor et. al., Wiley., Chapter 3. (Good reference).

The Fractal Geometry of Nature, Benoit Mandelbrot, W. H. Freeman 1983.

Random Number Generators, Birger Jansson, Victor Petterssons Bokindustrie Aktiebolag.

On the Meaning of Randomness and Some Ways of Achieving it, Martin Gardner, Scientific American, July, 1968, page 116

On the Problems of Random Vibrations, and of Random Flights in One, Two and Three Dimensions, Lord Rayleigh, The London, Edinburgh and Dublin Philosophical Magazine and Journal of the Series xxxi, April, 1919, p. 321

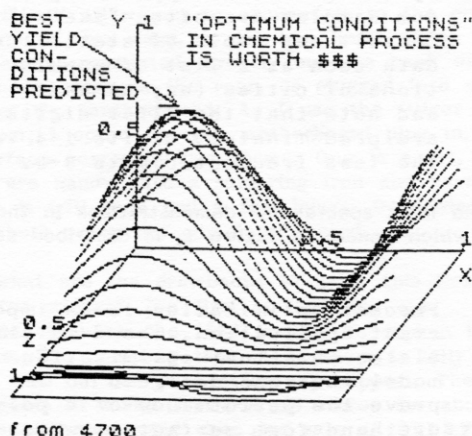


Figure 1. The computer in process control

SyncWare News

PIONEERS IFWARE

You have heard about hardware and software and SHAREware.

And SyncWare, of course.

Now meet IFWARE®.

We owe credit for this idea to Van Vangor, of Island Falls, Maine. We think it may help us to render you a real service. Here's how it works.

For some time, we have been considering the publication of an index to back copies of SyncWare News, as well as reprints of material we have published over the years. That is, IF there is enough demand to justify printing them.

And that's where the concept of IFWARE® comes in.

Here's the way it works. As a start, we propose to offer the following publications (at prices we still haven't determined--but we'll set them as low as we can. We hope to have prices by the time the next issue goes to press):

Index to volumes 1, 2, 3, and 4 of SyncWare News
BASIL's COMPENDIUM, Chapters 1-5, 6-10, 11-15

(NOTE: the COMPENDIUM will be upgraded, to take care of the TS2068 as well.)

All interested buyers will be asked to send a check for full payment along with their orders, accompanied by a stamped, self-addressed envelope. If within a month after publication of the offer we haven't received enough orders to pay for the cost of printing, we'll return the check in the SASE. If we do have enough orders, we'll return the SASE with the publication.

We'd appreciate your comments on this proposal, as well as any suggestions about other material you'd like to see reprinted.

TYD★BYTS

RECURSIVE PROGRAMMING

This little program is appropriate for early April. You probably have heard of the impressive power inherent in recursive programming techniques. This one makes it all clear.

```
10 PRINT "This program will teach you"
20 PRINT "all you want to know about"
30 GO TO 20
```


BASIL'S COMPENDIUM

TS1000 And TS2068

Basil Wentworth

That's right--Basil's Compendium is going to add the TS2068 to its repertoire. But you ZX81/TS1000 enthusiasts need not lose heart.

The two computers use the same microprocessor, the Z80, and are really not awfully different from each other. Sort of like brothers born a year or two apart.

And you won't have to worry about getting mixed up, either. The two brothers will continue to sleep in separate bedrooms, so to speak, so there won't be any danger of your not knowing which one is under consideration at the moment.

This installment, though, will be a sort of a REM statement in the Compendium series--it won't have any chapter number, and it will apply exclusively to the TS2068. After you have read it, you will be able to go back to the beginning of the Compendium and run the same exercises that you have been doing with the ZX81.

And what if you don't have the back issues of SyncWare News that published those exercises? Well, there's always the possibility that you can borrow them from your users' group library. Or you can buy the appropriate back issues. Volumes 2, 3, and 4 are available at \$16.95 each. (Each volume is 6 issues--one year's output.) Or all three for \$39.95. Postpaid in the United States. Send your orders to Jeff Moore, our publisher. His address is on our masthead.

If you do buy the back issues, you'll be getting some that don't contain the COMPENDIUM. On the other hand, you'll be getting a lot of other valuable material in addition to the COMPENDIUM.

Or you can wait for the reprint of the entire series to come out--upgraded to include the TS2068 from the beginning. We intend to publish the reprint on an IFWARE® basis--to be published IF there is enough demand. See elsewhere in this issue for the details.

One thing, though. The utility programs in the early chapters won't work on the TS2068. As a matter of fact, some of those utilities are not needed by the 2068, which has built into its operating system ways of doing the same things.

SO LETS GO

As far as machine coding is concerned, the 2068 differs from the 1000 series in three important respects: first, the "housekeeping" numbers are different; second, the address of the first program byte is different; and third, there are many more of what I described as "no-nos" in Chapter 7 (SWN Vol. 3 #3, page 16). So many of them, in fact, that it is a rare machine code program that won't evoke the report "K Invalid color" if you incorporate it in a 1 REM line.

There are ways to overcome this problem--one of the most useful is to tinker with bytes 23635 and 23636, which tell the computer where to start looking for the program--in essence, those bytes throw a camouflage net over the material at the bottom of the program, so the computer doesn't realize that it's there. But the camouflage is so effective that the computer won't SAVE or LOAD material down there, either.

Somewhere down the road, this series is going to include a whole section on locations for your code, both for the ZX81 and for the 2068. For the moment, though, let's make an arbitrary choice.

WHERE TO PUT THE CODE

Tell your computer to

```
PRINTUSR "a"
```

Be sure to include the quotation marks, or you may get in real trouble. (USR is invoked by pressing the "L" key after pressing CAPS SHIFT and SYMBOL SHIFT simultaneously. You'll be using it often.) You should get 65368. This marks the beginning of the area allotted to User Defined Graphics. Choose some base number smaller than 65368; for instance, 65000 is a very easy number to remember. A number even easier to use, 6e4 (=60000), is quite satisfactory if it's in an empty neighborhood.

Whatever base number you choose, remember it. Write it down, if you have a porous memory like mine. Then check to see whether it has already been used. RUN the following mini-program (as a direct command, without a line number):

```
FOR f=b TO b+100:PRINT f;" "; PEEK f: NEXT f
```

where b is the base number you have chosen. If you get all zeroes, you're OK. If that area is filled with an assortment of numbers, it is probably already in use (maybe your printer driver program is residing there, for instance), and you'd better choose another base number. Don't make it too small, though--and I can't tell you how small too small is. If you do choose too low a number, there's a danger of conflict with your program. And how small is too small depends, of course, on how much memory your program needs.

LOADING THE PROGRAM

For a beginner, the easiest way to build a machine code program on the 2068 is through the DATA/READ function. Something like the following:

```
9000 RESTORE
9010 LET m=65000
9020 READ a
9030 POKE m,a
9040 PRINT m;" ";PEEK m
9050 LET m=m+1
9060 GO TO 9020
```

and

```

9900 DATA 42,83,92,35
9910 DATA 17,10,0,1,10,0,62,13
9920 DATA 35,190,32,252,35,62
9930 DATA 35,190,200,190,216
9940 DATA 114,35,115,235,9,235
9950 DATA 24,235

```

This program, by the way, will renumber the lines of your program, according to the following rules:

Line 1, unchanged
 Subsequent lines renumbered 10,20...etc
 Lines 8960 and above, unchanged

And watch out for any GO TOs and GO SUBs--it's almost a sure bet that any lines they refer to will be changed.

THOSE BACK ISSUES

For your convenience, here is an index of the past chapters of BASIL's COMPENDIUM:

- 1 Introduction Vol. 2/1 p. 8
- 2 PEEK, POKE, and USR Vol. 2/2 p. 9
- 3 Where to Put the Program Vol. 2/3 p. 21
- 4 LET A=... Vol. 2/4 p. 15
- 5 Hexadecimal and 256-imal Vol. 2/5 p. 17
- 6 Deluxe Loader Program Vol. 2/6 p. 21
- 7 A Few No-nos Vol. 3/3 p. 16
- 8 LET A=...(again) Vol. 3/6 p. 10
- 9 Addition Vol. 4/1 p.13
- 10 Why Use Machine Code? Vol. 4/2 p. 20
- 11 GO TO/GO SUB Vol. 4/3 p. 19
- 12 IF...THEN Vol. 4/4 p. 22
- 13 FOR...NEXT Vol. 4/6 p. 18
- 14 Input/Output (PEEK and POKE) Vol. 5/1 p. 22

MEMOTECH CIF "COPY" FIX For Hot-Z II

Bill Bell
 TS1000 Machine code

Many TS-1000/ZX-81 users drive full size printers with the Memotech Centronics interface due to the wide availability and the ease of use with BASIC programming. However, machine code programs designed to operate the Timex printer have compatibility problems when producing screen copies that are longer or shorter than the standard 22 lines. A notable example is the single page copy function in HOT-Z II, which is a 24 line display.

Well, here is the explanation of the incompatibility, and a simple fix for HOT-Z II that allows EITHER printer to work with the changed code. The same trick can be applied to other programs experiencing the problem.

The Memotech CIF, besides having its own coding in the 8-10K region, also overrides and changes some of the original ROM code that controls printing.

Partial code without the interface:

```

0869 1616      LD D,16      ;22 line COPY
086B 2A0C40    LD HL,(DFIL) ;
086E 23        INC HL      ;display file
086F 1805      JR 0876      ;bypass LPRINT
0871 1601      LD D,01      ;1 line LPRINT
0873 213C40    LD HL,PRBF   ;printer buffer
0876 CDE702    CALL 02E7     ;call fast mode
0879 C5        PUSH BC      ;
087A E5        PUSH HL      ;
087B AF        XOR A        ;clear A register

```

Partial code with the interface attached:

```

0869 1616      LD D,16      ;22 line COPY
086B 2A0C40    LD HL,(DFIL) ;
086E 23        INC HL      ;display file
086F 1805      JR 0876      ;bypass LPRINT
0871 1601      LD D,01      ;1 line LPRINT
0873 213C40    LD HL,PRBF   ;printer buffer
0876 C5        PUSH BC      ;
0877 F5        PUSH AF      ;saves A reg
0878 7A        LD A,D        ;get line count
0879 FE16      CP 16         ;is it 22 lines?

```

```

087B 2004      JR NZ 0881    ;no,keep old A
087D F1        POP AF        ;yes,replace A
087E 3E76      LD A,76       ;with newline
0880 F5        PUSH AF       ;and resave

```

The changed code tests the D register for the number of lines. If it is 22, the routine replaces the A register content with a newline character in order to properly handle a COPY function. If it is not 22, it assumes we are doing an LPRINT and that a newline character was already in the A register from the LPRINT. This lack of newline is what causes our non-22 line COPY to fail. Easy to fix--all we have to do is load the A register with a newline (76 hex) before jumping into the copy routines. Now Memotech works OK, and the Timex ignores the A register.

The original HOT-Z II 64K program has the 24-line copy routine at 63F0 hex (A3F0 in a high RAM version) and looks like this:

```

63F0 1618      LD D,18      ;set D for
                                24 lines
63F2 2A0C40    LD HL,(DFIL) ;
63F5 23        INC HL      ;display file
63F6 CDE702    CALL 02E7     ;call fast mode
63F9 CD7608    CALL 0876     ;call copy screen
63FC C9        RET          ;return

```

Change (and simplify) the code to look like this:

```

63F0 1618      LD D,18      ;set D for
                                24 lines
63F2 3E76      LD A,76       ;put newline in A
63F4 CD6B08    CALL 086B     ;set HL and copy
63F7 C9        RET          ;return
63F8 00        NOP          ;unused
63F9 00        NOP          ;unused
63FA 00        NOP          ;unused
63FB 00        NOP          ;unused
63FC 00        NOP          ;unused

```

To make these changes permanent, use the method described in HOT-Z II documentation for creating a new version tape--the routine that ends up at 63F0 will be found at 4FC0 when loaded, or at 8FC0 if you read the tape into 8009 upwards.

Attention All Timex/Sinclair Users

MORE ON S.N.U.G.

LATEST RELEASE ON THE NATIONAL USERS' GROUP

The following news release was received from:

Mel Nathanson
7515 Arbordale Drive
Port Richey, FL 34668
tel 813 863-5552

He asks that correspondence be sent either to him or to:

Mary-Lynn Johnson
190 Hickory Woods Court, Unit 3C
Deltona, FL 32725

If anything in this release contradicts what we have already printed, this release, of course, takes precedence. If it duplicates something that we have already published, please forgive us--it probably merits saying twice, anyway.

S.N.U.G.--Sinclair Northamerica Users Group

During one of the organizational meetings of the recent Sunstate Timex/Sinclair Winterfest '88, the idea of a National organization for the advancement of Sinclair computing came up. It was decided that since we had developed a "core group" that was dedicated to promoting Sinclair computing, we would attempt to lay the groundwork for such an organization. It was also mentioned that the greater the amount of time from the departure of Timex from the computer industry, the less of an active market would result. Since we would have users from across the nation at the fest, it would be an ideal time to make our plans known. So, the Sinclair Northamerica Users Group, or SNUG (a name submitted to us by John Cushran, and later modified by Bill Jones) was starting to come closer to reality.

WHAT IT IS

The intent of SNUG is to provide a forum for the exchange of ideas. It would be a source of information, such as a listing of active members,

active Users Groups, Sinclair specific Bulletin Boards, an active library of Public Domain software, and a listing of available shareware and freeware. Later on we hope to propose an industry wide standard of hardware and software compatibility. So as not to have to reinvent the wheel, and to do this in the shortest amount of time, we are going to try to use an already established National group, such as CORSA (Corvair Owners Assn.) as a model to base our group on. SNUG would act as an umbrella Organization, with Regions being developed to tie in with established groups in those areas.

WHAT IT ISN'T

It is the intent of the organizers NOT to infringe or supersede any already established User Group or Vendor. It is intended to show some strength to the industry that Sinclair is not dead, and the mere fact that we can get this Organization together will prove that we can stick together and grow and prosper. We look on this as an enhancement to activities that have been planned on. Hopefully a Northamerican Calendar of events could be established to help co-ordinate any future plans and events. It is not designed to take anything away from anyone.

WHAT TO DO

We need the support of EVERY SINGLE SINCLAIR USER!

(editor's note: married users are welcome, too.) Whether you reside in Canada, the U.S., or Mexico, or for that matter anywhere, we need to know how you feel and what you want in this Organization. This is your opportunity to be heard...your comments, criticisms, complaints, or praises. What we have here is nothing more than an idea. Nothing at this time is set in concrete. We are more than open to suggestions. To make it work, these ideas of yours have to be forwarded immediately. We are putting on a time limit until June 30, 1988. If there is no support, then we will not proceed further. If there is input, we will update on a monthly basis to whoever will put the information in print. So, let's hear from you soon!

Applied Sinclair

Subroutines and Programs for the Mathematically Minded

Author Richard Booth
Sherman Fairchild Lab
Lehigh University
Bethlehem, PA 18015

Have you ever had some tricky or complicated calculations to make, or found yourself over your head with mathematic routines? If you have then this book is for you.

The 162 page book is packed with a wide variety of

useful subroutines for solving your most complex equations. There are 25 programs and 59 demonstrations to illustrate each principle. All are written in BASIC and compatible with all T/S computers. The programs and subroutines are written with economy in mind and leave a lot of empty space for your programs to use them.

The book is well organized in a building block fashion, from simple to very complex operations. The eight chapters and four appendixes cover such

topics as vectors, plotting, calculus, matrix operations, roots and regression, differential equations, and number representations. A scientific calculator program is given that operates similar to a standard calculator. There is an appendix that gives one line statements for operations not included in the T/S math functions.

Each program is well documented with a brief explanation of the math involved in the program, a "stat" sheet detailing the subroutines called, the variables used and changed, the listing, and a summary of the program. Each demonstration gives some test data, application and modification ideas, and encouragement for the user to experiment with his or her own ideas or data.

While the author states up front that the user may not understand all the mathematics contained in the book, well written instructions and suggestions for experimentation invite the user to explore and use

each demo until comfortable with each.

There are a few typographic errors, but these are very minor and easy to correct. The author has supplied an errata sheet that details all five of them and gives instructions to correct them.

This book is well worth the price just as a source book for solving those number crunchers or very precise calculations. If the user is a student of mathematics then this book can be a valuable asset. However, it is not necessary for you to know how they work, just how to use them.

Price \$13.00 for the book
\$16.00 for book and tape (2068)
for ZX81/1000/1500/2068

...reviewed by Kenneth Majors

Q Z O S C

T I M _ X
K/J
88

CRYPTOGRAMS ON THE COMPUTER

Basil Wentworth
TS2068 and TS1000
Machine code

Let me start out by saying what this program does NOT do. In spite of the title, it does not solve cryptograms. I enjoy working those little fellows far too much to surrender the fun to a computer. This program is designed, rather, to do the clerical jobs for me, while I save the creative work for myself.

If you have ever worked cryptograms, you know that the solutions are sometimes obvious. If, for example, the encrypted text begins like

BA BI ASLO AVEA EJJ...

you can tell, almost by intuition, that the solution is going to start out

IT IS TRUE THAT ALL...

Other patterns are easy to recognize. JOCJEO is very likely to be PEOPLE, unless the word before it is EO, in which case it is probably INDIAN. BCDBEF will be ALWAYS. Well, almost always.

But the majority of crypto texts take more work than that. At least, for amateurs like me. If they didn't, it wouldn't be any fun. That single letter, for example: is it I or A? Is that BCC going to be ALL or SEE or OFF or ASS or TOO, or what? If there's a BC somewhere in the exercise, you can (probably) eliminate ALL and SEE, but the other three are still possible. So are BEE and INN and WEE. There may be others. At any rate, you're likely to run into a lot of trial-and-error work, leaving the page pretty well scrubbed up by your eraser. This program lets you do the erasing on your screen. Here's how it works.

First, you are told to enter the program in 32-character lines. If you try to make the line longer than that, the computer will just chop it back to 32. There has been an unfortunate lack of co-ordination between Sir Clive and most cryptogram publishers in this area, as the magazines almost always print lines a little longer than 32 characters. We'll just have to live with their lack of foresight.

When you have the 32 characters ready, press "ENTER", and start on the next line. When all of the lines have been typed in, signal this fact by pressing an extra "ENTER". The computer will then print the lines of text, triple-spaced, with a "<" and a ">" indicating the positions of the plain text (the part you can read) and the encrypted text. These symbols are vital to the operation of the machine code part of the program. (I used a slightly different system for the TS1000, due to the straightforward simplicity of that computer's display file.)

As you can see, the plain text (which is blank for the moment) will appear above the encrypted version.

At the same time, a prompt at the bottom of the screen will ask for the OLD LETTER. Now it's your turn to do some work. Look at the encrypted text, put your brain in gear, and figure out one substitution. In the example above (BA BI ASLO etc), you expect to replace B with I. The "old" letter, then, is B. Press B. You will get a new prompt, asking for the NEW LETTER--in this case, I. So press I. Immediately, all of the B's in the encrypted text have I's printed above them. And the prompt again asks for an "old" letter. If you change your mind about the B's, press B again, followed by your new choice. Or followed by a space, if you want to blank out the letters above

the B's. On the other hand, if you're satisfied with your choice for B, work on another letter.

If you garble things so badly that you want to start over, press "ENTER" in lieu of an OLD letter. With the 2068, the program then gives you a chance to reprint the encrypted text (by pressing Y) or to quit that cryptogram entirely (by pressing any other letter). With the TS1000 program, this response gives you a STOP. If this makes you unhappy, you can easily modify the program to emulate the 2068 routine.

If you choose not to repeat (2068), you can either start at the beginning for a new cryptogram (press Y) or quit entirely (press any other letter).

The Machine Code Section

I should mention here a warning that applies whenever you are using machine code: DON'T try to use the same memory bytes to hold more than one machine code routine. That should be obvious, but I found myself making just that mistake, when I tried to LOAD my printer software into the same memory locations that I was using for the CRYPTO bytes. It didn't work.

Before I go into a discussion of the machine code section, permit me one short digression. That is to point out that line 10 makes sure that the computer is in the CAPS mode--that all the letters entered from the keyboard will be capitals, following the convention of printed cryptograms.

The machine code for the TS 2068, which is listed in Figure 1, is quite straightforward. The code is stored above RAMTOP by lines 9300 on, in bytes beginning with 65280. You will notice that the USR call (line 3260) is to byte 65290, and the bytes from 65280 to 65287 are taken up by a subroutine, with two nulls at 65288 and 65289.

The machine code program first searches the VARS area of the computer memory to find where "<" (the beginning of the string of encrypted text) is stored, and records this number in HL. Then it goes through a similar procedure for the string of plain text (beginning with ">"), and records that number in DE. The length of the line being worked on is held in BC. From then on, it is a simple matter of working through the two strings, byte by byte (the duration of the procedure governed by the bytes beginning with 65315) and POKEing the designated "new" character into the plain text string every time the selected "old" character is encountered in the encrypted string.

The TS1000 machine code works a little differently, POKEing the appropriate data directly into the display file at the proper place.

And there you are. Here is one of my limericks to get you started.

BA BSI BL LOI VEE JBCY "NELOIT,
DI JFTIGH WBA LTFJL IBWO ELOIT:
HEF DETTH JE NFWO
BXEFL YBTDCA BAY JFWO--
BN C LOIA NH UIISIT'J XTELOIT?"

Those of you who are as inexperienced as I am about machine code might be interested in the "rough draft" I wrote for the TS1000 routine, simulating in BASIC the m/c operation.

Memory	Byte	Mnemonic	Remarks
65280	42	LD HL,	Vars area
1	75	(23627)	
2	92		
3	35	INC HL	Look for a
4	190	CP(HL)	match
5	200	RET Z	
6	24	JR,-5	
7	251		
8	0	NOP	nulls
9	0		
65290	62	LDA,	USR to here
1	62	62	(<)
2	205	CALL 65280	
3	0		
4	255		
5	235	EX DE,HL	
6	62	LD A,	
7	60	60	(>)
8	205	CALL 65280	
9	0		
65300	255		
1	1	LD BC,	LEN A\$, by
2	x		lines 3000 and
3	x		3010
4	62	LD A,	Old ltr, by
5	x		3190
6	190	CP(HL)	
7	32	JR NZ, 4	
8	4		
9	235	EX DE,HL	
65310	54	LD(HL),	New ltr, by
1	x		3240
2	235	EX DE,HL	
3	35	INC HL	
4	19	INC DE	
5	11	DEC BC	
6	62	LD A, 0	
7	0		
8	184	CP B	
9	32	JR NZ,-17	
65320	239		
1	185	CPC	
2	200	RET Z	
3	24	JR,-21	
4	235		

Figure 1. The 2068 machine code

```

1 POKE 23658,8
10 GO TO 9000
1000 PAUSE 10
1005 IF INKEY#="" THEN GO TO 100
S
1010 RETURN
1500 CLS
1510 PRINT ""ENTER ENCRYPTED TE
XT, ONE LINE"
1520 PRINT "AT A TIME. THE COMP
UTER WILL"
1530 PRINT "LOP OFF EACH LINE TO
32 CHARAC-"
1540 PRINT "TERS IN LENGTH."
1550 PRINT ""WHEN YOU HAVE TYPE
D IN ALL THE"
1560 PRINT "TEXT, PRESS N/L"
1570 PRINT ""READY? PRESS N/L
TO BEGIN"
1580 INPUT A$
1590 CLS
2000 LET A#=""<"
2010 LET B#="">"
2020 LET N=31
2030 INPUT T$
2040 IF T#="" THEN GO TO 3000
2050 LET T#=(T#+
) ( TO N)

```

```

2060 LET U$=""
2070 LET B$=B$+U$
2080 LET A$=A$+T$
2090 LET N=32
2100 PRINT "T$
2110 GO TO 2030
3000 POKE 65303,INT (LEN A$/255)
3010 POKE 65302,LEN A$-255*PEEK
65303
3100 CLS
3110 LET I=INT (LEN A$/32)
3120 FOR F=0 TO I-1
3130 PRINT B$(32*F+1 TO 32*(F+1)
)
3140 PRINT A$(32*F+1 TO 32*(F+1)
)
3150 NEXT F
3160 PRINT AT 19,0;"OLD LETTER?"
-
3170 GO SUB 1000
3180 IF INKEY$=CHR$ 13 THEN GO T
O 3500
3190 POKE 65305,CODE INKEY$
3200 PRINT INKEY$;" ";
3210 PRINT "NEW LETTER - ";
3220 GO SUB 1000
3230 IF INKEY$=CHR$ 13 THEN GO T
O 3500
3240 POKE 65311, CODE INKEY$
3250 PRINT INKEY$
3260 RANDOMIZE USA 65290
3270 CLS
3280 GO TO 3110
3500 CLS : PRINT "TRY THIS CRY
PTOGRAM AGAIN? Y- N-"
3510 GO SUB 1000
3520 IF INKEY$<>"Y" AND INKEY$<>
"Y" THEN GO TO 3550
3530 LET B$=">"
3540 FOR F=1 TO LEN A$-1: LET B$
=B$+" " : NEXT F
3550 GO TO 3100
3560 CLS : PRINT "TRY ANOTHER
CRYPTOGRAM? Y- N-"
3570 GO SUB 1000
3580 IF INKEY$="Y" OR INKEY$="Y"
THEN GO TO 1500
3590 CLS : PRINT "TS 2063 THAN
KS YOU; BASIL LOVES"
3600 PRINT "JOCELYN"
3610 PRINT "BIBI"
9999 STOP
9000 CLEAR 65270
9010 LET F=65279
9020 LET F=F+1
9030 READ A: IF A=64 THEN RESTOR
E : RUN 1500
9040 POKE F,A: GO TO 9020
9050 DATA 42,75,92,35,190,200,24
,251,0,0,62,62,205,0,255,235
9060 DATA 62,60,205,0,255
9070 DATA 1,0,0,62,0,190,32,4,23
5,54,0,235,35,19,11
9080 DATA 62,0,184,32,239,185,20
0,24,235
9400 DATA 64
9499 STOP
9500 SAVE SAVE
9510 SAVE "CRYPTOLIST" LINE 1
9520 BEEP .3,12: BEEP 1,12
9530 VERIFY "CRYPTOLIST"
9540 BEEP .3,16: BEEP 1,16

```

CRYPTO--TS2063 LISTING

```

.....
1 REM ERAND-4* INKEY$ =7+ASN
RNDY<COS Y$TAB RAND*5 GOSUB P
IO=>RAND
10 GOTO 1000
100 INPUT T$
110 IF T$="" THEN LET R=1
150 LET T$=(T$+"") ( TO 32)
160 RETURN
999 STOP
1000 LET L=2
1010 LET R=0
1100 PRINT AT 19,0;"ENTER ENCRYP
TED TEXT, 32 LETTERS AT A TIME.";
1105 PRINT " (LESS THAN 6 LINES.
)";
1110 PRINT " PRESS ""ENTER"" TO
STOP"
1120 GOSUB 100
1130 IF R THEN GOTO 2000
1140 PRINT AT L,0;T$
1150 LET L=L+3
1160 GOTO 1020

```

```

2000 PRINT AT L,0;"<"
2010 PRINT AT 19,0;"INPUT OLD LE
TTER -"
2015 PRINT "
"
2020 PRINT "PRESS ""ENTER"" TO S
TOP"
2030 INPUT L$
2050 IF L$="" THEN GOTO 5000
2060 POKE 16533,CODE L$
2110 PRINT AT 19,0;"INPUT NEW LE
TTER -"
2130 INPUT L$
2150 IF L$="" THEN GOTO 5000
2160 POKE 16544,CODE L$
2500 PRINT USA 16514
2510 GOTO 2010
9999 STOP
9000 REM SAVE
9010 CLS
9020 PRINT AT 15,0;"START RECORD
ER; THEN PRESS ANY KEY"
9030 PAUSE 4E4
9100 SAVE "CRYPTO"
9110 CLS
9120 RUN

```

CRYPTO--TS1000 BASIC PROGRAM

```

.....
.....
16514 42      LD HL,
16515 12      (16396)
16516 64
16517 22      LD D,
16518 32      32
16519 1       LD BC,
16520 65      65
16521 0
16522 9       ADD HL,BC
16523 35      INC HL
16524 21      DEC D
16525 202     JP Z,
16526 133     16517
16527 64
16528 62      LD A,
16529 19      19
16530 190     CP(HL)
16531 200     RET Z
16532 62      LD A,
16533 49      49*
16534 190     CP(HL)
16535 194     JNZ,
16536 139     16523
16537 64
16538 1       LD BC,
16539 33      33
16540 0
16541 237     SBC HL,BC
16542 66
16543 54      LD(HL),
16544 61      61**
16545 9       ADD HL,BC
16546 195     JP,
16547 139     16523
16548 64

```

* OLD LETTER
** NEW LETTER

CRYPTO--TS1000 MACHINE CODE


```

10 PRINT
20 PRINT
30 PRINT "BASIL LOVES JOCELYN."
40 PRINT
50 PRINT
60 PRINT "BASIL LOVES JOCELYN."
70 PRINT
80 PRINT
90 PRINT "<"
100 LET HL=PEEK 16396+256*PEEK
16397
110 LET D=32
120 LET BC=65
130 LET HL=HL+BC
140 LET HL=HL+1
150 LET D=D-1
160 IF D=0 THEN GOTO 110
170 LET A=19
175 REM CODE "<"
180 IF PEEK HL=A THEN STOP
190 LET A=CODE "L"
195 REM OLD LETTER
200 IF PEEK HL<>A THEN GOTO 140
210 LET BC=33
220 LET HL=HL-BC
230 POKE HL, CODE "X"
235 REM NEW LETTER
240 LET HL=HL+BC
250 GOTO 140

```

"ROUGH DRAFT" OF CRYPTO--TS1000
PROGRAM

THE CATS MEOW

Software review
TS2068

By Basil

One of the souvenirs I brought back from the WINTERFEST was a copy of the CATS library of public domain programs. In case you don't know, CATS is the Capital area (Washington DC) users' group. The library now consists of 11 volumes, on 5 cassettes. The list is growing.

Inquiries should be sent to

John Riley
120 North Fairlawn Dr.
Carrollton, GA 30117

The Georgia address must make CATS about the most spread-out users' group in existence, in terms of geography. The "home office" is

P.O. Box 467
Fairfax Station, VA 22039

I haven't begun to try all of the programs, but here is a fast run-down on the catalog.

- Vol. one--games
- Vol. two--educational programs and utilities
- Vol. three--more utilities
- Vol. four--SPECTRUM programs
- Vol. five--more 2068 utilities
- Vol. six--a mixed bag. (I notice that one of their games is called GHOTI, which should endear them to disciples of George Bernard Shaw.)
- Vol. seven--more SPECTRUM software
- Vol. eight--more utilities
- Vol. nine--graphics
- Vol. ten--education and demos
- Vol. eleven--graphics and character sets

As I say, I've just begun to scratch the surface. However, here are a few samples of my favorite text, printed from programs in Volume 11:

Standard 2068
Basil loves Jocelyn

Big Letters
Basil loves Jocelyn

Alternate Characters
Basil loves Jocelyn

BOLD--Basil loves Jocelyn

crazyprint--Basil loves Jocelyn

Print--Basil

loves Jocelyn

~~Print--Basil loves Jocelyn~~

Etc.

3D Words
Basil
loves
Jocelyn

Old English
Basil loves
Jocelyn

ETAMITLU - - The Simplest Word

Processor

by Basil
TS1000 or TS2068
Some machine code

Don't ask me how to pronounce that word. Don't ask what it means, either. If you know how my mind works, you won't have to ask. If you don't--well, you remember what Louie Armstrong said about jazz: "If you have to ask what it is, you wouldn't understand the answer."

Be that as it may, this is the simplest word processor of all. Don't be afraid to use it, if you have an article that is just begging to be written but you don't have MEMOTEXT or MSCRIPT.

The program is simplicity itself. Figure 1 LISTs the 2068 program, with an abridged version of this article in it. Figure 2 is the "empty" TS1000 version.

As you can see, the text is stored as successive lines of string variable, designated as a\$. The operation of the SAVE, PRINTOUT, and DISPLAY sections should be apparent. The DISPLAY routine lists the line numbers, as well as the text, to make it easy to locate any errors that are detected.

Line 9160 determines which lines will be printed or displayed.

Just a word about the machine code section. This is a simple line renumbering sequence, established by DATA/READ statements in the 2068, and residing in the first REM line in the TS1000 version. It serves the two functions of POKEing zero as the number of the first line, to protect it from inadvertent erasure, (actually, this is done by Line 9510 of the TS1000 version) and of letting you tidy things up if you want to insert or DELETE lines as your article takes shape. Just RUN the program after you make the insertion or deletion, and everything "neatens up" as pretty as my granddaughter in her Sunday clothes.

If you use this processor to prepare an article for SyncWare News, remember that our format is 52 characters per line. For your information, a 52-character line looks like this in the LISTing:

```
xxxx LET a$="1234567890123456789
01234567890123456789012345678901
2"
```

The program does not renumber Line 2 (Line 5 in the TS1000 version). Also it leaves lines numbered higher than 8959 (=256*35-1) unchanged.

```
9030 PRINT "DO YOU WANT"
9040 PRINT "DISPLAY?"
9050 PRINT "PRINTOUT?"
9070 PRINT "PRESS INITIAL
LETTER)"
9080 SLOW
9100 IF INKEY$="" THEN GOTO 9100
9110 FAST
9120 LET C=CODE INKEY$
9130 IF C<>41 AND C<>53 THEN GOT
O 9080
9135 CLS
9140 FOR F=10 TO 50 STEP 10
9150 GOSUB F
9160 IF C=41 THEN GOTO 9200
9170 LPRINT A$
9180 GOTO 9210
9200 PRINT F;" ";A$
9210 NEXT F
9300 STOP
9500 SAVE "ETAMITLU"
9510 POKE 16510,0
9520 LIST
```

TS1000 ETAMITLU LISTING

```
16514 1
16515 5
16516 0
16517 33
16518 129
16519 64
16520 62
16521 117
16522 60
16523 35
16524 190
16525 32
16526 252
16527 35
16528 190
16529 200
16530 62
16531 33
16532 190
16533 248
16534 112
16535 35
16536 113
16537 229
16538 33
16539 5
16540 0
16541 0
16542 68
16543 77
16544 225
16545 195
16546 136
16547 64
```

MACHINE CODE FOR TS1000 ETAMITLU
(SEE BASIL'S COMPENDIUM CHAPT. 2
IN SYNCWARE NEWS, VOL. 2, NO.2)

```
1 REM "S. RANDY?W74 UNPLOT 7
203 YES SAVE ??? FAST 51 ??? LP
PRINT ?AND
5 GOTO 9000
10 LET A$="THE WORLD KNOWS THA
T"
15 RETURN
20 LET A$="BASIL"
25 RETURN
30 LET A$="LOVES"
35 RETURN
40 LET A$="JOCELYN"
45 RETURN
8999 STOP
9010 RAND USR 16514
9020 CLS
```

```
1 GO TO 9000
2 LET a$="": RETURN
10 LET a$="": RETURN
20 LET a$="": RETURN
30 LET a$="": RETURN
40 REM lines 40 to....
620 REM ...620
630 LET a$="": RETURN
640 LET a$="The program does no
t renumber Line 2 (Line 5 in the
": RETURN
650 LET a$="TS1000 version). A1
```



```

so it leaves lines numbered high
er": RETURN
660 LET a$="than 8959 (=256*35-
1) unchanged.": RETURN
670 LET a$=CHR$ 12: RETURN
680 LET a$="": RETURN
8998 RETURN
8999 STOP
9000 REM LPRINT and PROOFREAD
9010 RANDOMIZE USR 65000
9020 CLS
9030 PRINT ""Do you want"
9040 PRINT "" Display?"
9050 PRINT "" Printout?"
9060 PRINT "" Edit or Enter tex
t?"
9070 PRINT ""(Press initial let
ter)"
9080 IF INKEY$<>" THEN GO TO 9
080
9090 IF INKEY$="" THEN GO TO 90
90
9100 LET C=CODE INKEY$
9110 LET C=C*((C=68)+(C=69)+(C=8
0)+(C=100)+(C=101)+(C=112))
9120 IF C=0 THEN GO TO 9080
9130 CLS
9140 LET C=C-32*(C>80)
9150 IF C=69 THEN LIST
9160 LET P=(C=80)
9170 FOR F=20 TO 670 STEP 10
9180 GO SUB F
9190 IF P THEN LPRINT a$: GO TO
9210

```

```

9200 PRINT F;" ";a$
9210 NEXT F
9550 BEEP .3,12: BEEP 1,12
9799 STOP
9800 REM SAVE
9805 POKE 23636,104
9810 SAVE "ETAMITLU" LINE 9860
9820 BEEP .3,12: BEEP 1,12
9830 VERIFY ""
9840 BEEP .3,16: BEEP 1,16
9850 STOP
9860 CLEAR 65000: REM if varied,
change lines 9010 and 9875 acco
rdingly
9865 POKE 23636,105: POKE 26802,
251: RANDOMIZE USR 26921
9870 RESTORE
9875 FOR f=65000 TO 65036
9880 READ a
9890 POKE f,a
9900 NEXT f
9910 BEEP .3,12: BEEP 1,12
9920 DATA 42,83,92,35,54,0,1,25,
0
9930 DATA 9,17,10,0,1,10,0,62,13
9940 DATA 35,190,32,252,35,62,35
9950 DATA 190,200,190,216,114,35
,115,235,9
9960 DATA 235,24,235

```

2068 LISTing for ETAMITLU

STITCHES IN TIME

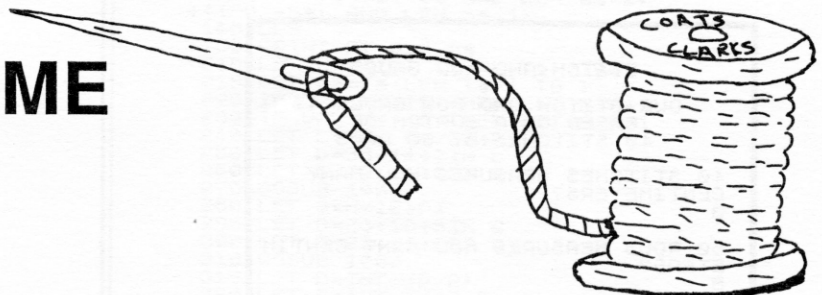
Dorothea Bundy
Contest entry
2068 or 1000, BASIC

A programmer I am not, but a knitter I am. One of the routines in knitting has been reduced to a minimum with the help of a knitting machine. Now what takes longer than the actual knitting of an article is figuring out the pattern and adjusting it to the specific measurements of the individual. This conversion is a routine, or rather combination of routines which I have made one of my knitting/computer goals. "GAUGE" and "GAUGECHART" are two of these routines.

Any knitter, whether male or female, knows that you don't just pick up knitting needles, yarn and pattern and begin knitting without first figuring out your stitch and row gauge for the specific yarn and needles that you are using. You knit a sample swatch, block it, measure 40 stitches by 60 rows of it in centimeters and figure your gauge.

"GAUGECHART" prints out your stitch and row gauges in chart form; "GAUGE" gives the same result but is the routine I have written for use within my knitting program. It requires input of the stitch and row measurements of the worked swatch.

Agreed, probably not too many computer nuts are



interested in knitting, but I'm sure that I am not alone, and I have seen no programs at all geared to help knitters.

"FLOWERS" was a fun thing that would look better 6 to 8 times larger but would probably be entirely too long in BASIC.

The "GAUGECHART" gives a voluminous printout--over a foot and a half on the TS2040 printer. To give you an idea of its output, here are the first few and the last few lines:

CM	INCH	ST	ROW
10	4	10.16	15.24
10.1	4.04	10.05	15.08
.	.	.	.
21.8	8.72	4.66	6.99
21.9	8.76	4.63	6.95

The "GAUGE" program doesn't take nearly as much paper. Its printout is shown in the illustration.

In this case, the numbers 3 and 5 were supplied by your Editor, who has no idea whether or not they are realistic. The calculated results, 33.87 stitches per inch and 30.48 rows per inch, are accurate, but are likely to be pure GIGO.

EDITOR's note: Programmer she may not be, but Miss Bundy certainly has a fine instinct for matching the abilities of the computer with the needs of real life. There must be a lot of our readers who have found ways to use their computers in their other hobbies. Let's hear from you.

```

300 CLS : PRINT TAB 6;"STITCH A
ND ROW GAUGE": PRINT
310 PRINT TAB 2;"OUR STITCH AN
D ROW GAUGE IS" TAB 6;"BASED ON
A SWATCH OF" TAB 4;" 40 STITCHES
BY 60 ROWS": PRINT
320 REM 40 STITCHES/C * 2.54=ST
ITCHES PER 1"
325 REM 60 ROWS/CM * 2.54=ROWS
PER 1"
330 PRINT "40 STITCHES MEASURES
HOW MANY CENTIMETERS?"
340 INPUT C: PRINT C: PRINT
350 PRINT "60 ROWS MEASURES HOW
MANY CENTIMETERS?"
360 INPUT CM: PRINT CM: PRINT
370 LET ST=40/C*2.54: LET R=60/
CM*2.54
380 PRINT TAB 2;"STS PER INCH",
"ROWS PER INCH": PRINT
390 PRINT TAB 6;INT (ST*100+.5
)/100 TAB 21;INT (R*100+.5)/100
8999 STOP
9000 SAVE "GAUGE" LINE 0

```

LISTING of "GAUGE"

```

          STITCH AND ROW GAUGE

    OUR STITCH AND ROW GAUGE IS
    BASED ON A SWATCH OF
    40 STITCHES BY 60 ROWS

40 STITCHES MEASURES HOW MANY
CENTIMETERS?
3

60 ROWS MEASURES HOW MANY CENTIM
ETERS?
5

    STS PER INCH    ROWS PER INCH

        33.87          30.48

```

PRINTOUT of "GAUGE"

```

100 CLS
110 REM **GAUGE CHART
130 LET CM=10.0: LET I=CM*.4:
LET ST=40/CM*2.54: LET R=60/CM*2
.54
135 PRINT TAB 5;"KNITTING GAUGE
CHART"
140 PRINT TAB 1;"CM";TAB 9;"INC
H";TAB 16;"ST";TAB 24;"ROW"
145 FOR C=9 TO 21 STEP 0.1
150 PRINT TAB 1;CM;TAB 9;I;TAB
16;ST;TAB 24;R
155 LET CM=CM+.1: LET I=INT (C
M*.4*100+.05)/100: LET ST=INT
(40/CM*(2.54)*100+.05)/100: LET
R=INT (60/CM*(2.54)*100+.05)/1
00
160 NEXT C
8999 STOP
9000 SAVE "GAUGECHART" LINE 0

```

LISTING of "GAUGECHART"

```

1 CLS
10 FOR X=0 TO 7: READ N: POKE
USR "A"+X,N: NEXT X
20 DATA 0,0,0,0,0,0,0,0
30 FOR X=0 TO 7: READ N: POKE
USR "B"+X,N: NEXT X
40 DATA 0,0,0,0,1,1,2,2
50 FOR X=0 TO 7: READ N: POKE
USR "C"+X,N: NEXT X
60 DATA 0,64,64,160,80,16,8,8
70 FOR X=0 TO 7: READ N: POKE
USR "D"+X,N: NEXT X
80 DATA 0,0,0,0,0,0,0,0
90 FOR X=0 TO 7: READ N: POKE
USR "E"+X,N: NEXT X
100 DATA 0,0,0,0,0,0,0,0
110 FOR X=0 TO 7: READ N: POKE
USR "F"+X,N: NEXT X
120 DATA 0,0,0,1,9,60,127,64
130 FOR X=0 TO 7: READ N: POKE
USR "G"+X,N: NEXT X
140 DATA 2,2,194,226,51,115,99,
254
150 FOR X=0 TO 7: READ N: POKE
USR "H"+X,N: NEXT X
160 DATA 8,8,72,250,89,187,171,
218
170 FOR X=0 TO 7: READ N: POKE
USR "I"+X,N: NEXT X
180 DATA 0,0,192,224,100,143,25
3,192
190 FOR X=0 TO 7: READ N: POKE
USR "J"+X,N: NEXT X
200 DATA 0,0,0,0,0,0,128,128
210 FOR X=0 TO 7: READ N: POKE
USR "K"+X,N: NEXT X
220 DATA 129,131,196,4,9,8,8,4
230 FOR X=0 TO 7: READ N: POKE
USR "L"+X,N: NEXT X
240 DATA 255,198,89,255,181,10,
5,2
250 FOR X=0 TO 7: READ N: POKE
USR "M"+X,N: NEXT X
260 DATA 221,156,211,255,85,170
,84,168
270 FOR X=0 TO 7: READ N: POKE
USR "N"+X,N: NEXT X
280 DATA 208,184,68,228,178,2,6
,0
290 FOR X=0 TO 7: READ N: POKE
USR "O"+X,N: NEXT X
300 DATA 64,64,192,0,0,0,0,0
310 FOR X=0 TO 7: READ N: POKE
USR "P"+X,N: NEXT X
320 DATA 6,0,0,0,0,0,0,0
330 FOR X=0 TO 7: READ N: POKE
USR "Q"+X,N: NEXT X
340 DATA 1,1,2,2,3,6,15,0
350 FOR X=0 TO 7: READ N: POKE
USR "R"+X,N: NEXT X
360 DATA 80,80,168,168,248,172,
254,0
370 FOR X=0 TO 7: READ N: POKE
USR "S"+X,N: NEXT X
380 DATA 0,0,0,0,0,0,0,0
390 FOR X=0 TO 7: READ N: POKE
USR "T"+X,N: NEXT X
400 DATA 0,0,0,0,0,0,0,0
410 FOR X=2 TO 16
420 PRINT AT 2,2;"B";"C"
430 PRINT AT 3,1;"F";"G";"H";"I
";"J"
440 PRINT AT 4,1;"K";"L";"M";"N
";"O"
450 PRINT AT 5,1;"P";"Q";"R";"S
";"T"
500 STOP
9000 SAVE "FLOWERS" LINE 0

```

NOTE: Lines 420 TO 450 print graphics CHR\$S

```

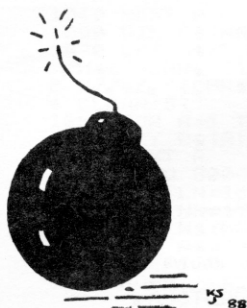
420 PRINT AT 2,2;" / " " / "
430 PRINT AT 3,1;" / " " / " " / "
440 PRINT AT 4,1;" / " " / " " / "
450 PRINT AT 5,1;" / " " / " " / "

```

LINES 420 to 450 AFTER user-defined graphics characters have been generated



PRINTOUT of "FLOWERS"



"TIME" BOMB

Bruce Shideler
TS1000, BASIC
A contest entry

This classroom project teaches the students to tell time--assuming that there will still be a need for this archaic skill in the era of digital watches.

The student is confronted with a series of clock faces, with the hands at randomly-selected positions. The complexity of the time indicated by the clock depends on the difficulty index, which is set at the beginning of the session.

Somewhere backstage, a time bomb reads 5 seconds, and counting. Fortunately, the counting is not in real time, as this is a rather slow-running program. Every time the student gives the time correctly, one second is added to the bomb's life; on the other hand, the apocalypse is brought closer by one second when a wrong answer is given. If the detonation time is expanded to 10 seconds, the bomb is considered to be de-fused; if it gets down to 0, then...

The success and failure messages, given in lines 2010 and 2110, are models of economy.

Editors Comments

As I mentioned above, this is a slow-running program, although Mr. Shideler has taken advantage of a few programming techniques to reduce the delay. One clever stratagem is to store the clock face, once it has been generated, as A\$, so that a simple "PRINT A\$" rapidly fills the screen with the numbers and index marks of the clock. The hands, of course, have to be put in separately, as they must appear in various positions.

Another wise move is to identify the hour hand with an inverse "S", so that even the youngest (or the oldest) eyes will have no difficulty telling the two hands apart.

I have a couple of questions and comments, however. First, I don't see the need for the long delay at line 3. Second, the subroutine at 3000 is a real time-killer, and the students would appreciate some sort of courtesy message, like "Don't go 'wa-a-ay!'" to warn them that they haven't been deserted. Third, since the clock face is good forever, once it has been generated, it would be helpful to make provision for another run without having to go through the wait all over again. And fourth, isn't A\$ SAVED along with the rest of the program? Couldn't the delay be avoided entirely by having the program automatically GOTO 10?

Come to think of it, it would probably be even better (safer, anyway) to store the clock face in a PRINT statement, by the technique presented in the forewords to BASIL's COMPENDIUM, Chapters 8 and 9 (SyncWare News, Vol. III number 6 and Vol. IV number 1).

```

1 REM "TIME BOMB"
2 PRINT "TIME BOMB"
3 PAUSE 500
4 CLS
5 FAST
6 GOSUB 3000
7 SLOW
8 CLS
9 PRINT "YOU HAVE FIVE SECOND
S TO DEFUSE A BOMB. YOU WILL BE
SHOWN A SERIES OF CLOCK FACE
S SHOWING TIMES PICKED AT RAND
OM. IF YOU ENTER THE CORRECT TI
ME YOU ARE GIVEN AN EXTRA SECON
D. IF YOU ENTER THE WRONG TIME
IT COSTS YOU A SECOND. IF TH
E COUNT REACHES ZERO THE BOM
B EXPLODES. IF THE COUNT REACHES
10 YOU DEFUSE THE BOMB. DO
NOT ENTER A COLON TO ENTER THE
TIME. 1:30 IS ENTERED AS 130.
SELECT LEVEL (1-5)"
10 INPUT LEVEL
11 LET COUNT=5
12 LET T=INT (RND*12)+1
13 LET T$=STR$ T
14 GOTO (90 AND LEVEL=1)+(100
AND LEVEL=2)+(110 AND LEVEL=3)+(
120 AND LEVEL=4)+(130 AND LEVEL=
5)
90 LET T$=T$+"00"
95 GOTO 140
100 LET T=INT (RND*2)*30
102 LET S$=STR$ T
105 LET T$=T$+(S$ AND LEN S$=2)
+((("0"+S$) AND LEN S$=1)
106 GOTO 140
110 LET T=INT (RND*4)*15
115 LET S$=STR$ T
116 LET T$=T$+(S$ AND LEN S$=2)
+((("0"+S$) AND LEN S$=1)
117 GOTO 140
120 LET T=INT (RND*12)*5
125 LET S$=STR$ T
126 LET T$=T$+(S$ AND LEN S$=2)
+((("0"+S$) AND LEN S$=1)
127 GOTO 140
130 LET T=INT (RND*60)
135 LET S$=STR$ T
136 LET T$=T$+(S$ AND LEN S$=2)
+((("0"+S$) AND LEN S$=1)
140 CLS
150 PRINT AT 0,0;A$
210 LET H=2*INT ((VAL T$)/100)
215 LET M$=T$(LEN T$-1 TO )
220 IF VAL M$>15 THEN LET H=H+1
230 IF VAL M$>45 THEN LET H=H+1
240 LET L=VAL M$/30*PI
250 LET D=33+14*SIN L
260 LET E=22+14*COS L
270 GOSUB 1000
280 LET G=H/12*PI
290 LET D=33+10*SIN G
300 LET E=22+10*COS G
310 GOSUB 1000
315 LET D=INT (D/2)
320 LET E=INT ((43-E)/2)
325 PRINT AT E,D;"S"
330 PRINT AT 0,25;COUNT
340 INPUT R$
350 IF R$<>T$ THEN GOTO 385
355 PRINT AT 21,0;R$;" IS CORRE
CT"
360 LET COUNT=COUNT+1
370 IF COUNT=10 THEN GOTO 2000
380 GOTO 410
385 PRINT AT 21,0;"WRONG, CORRE
CT TIME: ";T$
390 LET COUNT=COUNT-1
400 IF COUNT=0 THEN GOTO 2100
410 GOTO 40
1000 REM DRAW
1010 LET X=33
1020 LET Y=22
1050 LET A=X-D
1060 LET B=Y-E
1070 LET C=(A AND ABS A>ABS B)+(
B AND ABS B>ABS A)
1080 IF C=0 THEN LET C=0.1
1090 FOR F=0 TO C STEP SGN C
1100 PLOT X+A/C*-F,Y+B/C*-F
1110 NEXT F
1120 RETURN
2000 CLS
2010 PRINT "YOU DID IT!";"YOU DEFE
ATED THE BOMB";
2020 GOTO 2010
2100 CLS
2110 PRINT "BOOM!";"BOOM!";
2120 GOTO 2110
3000 REM SET UP CLOCK FACE
3010 FOR D=1 TO 12
3020 PRINT AT 10-8*COS (D/6*PI),
16+8*SIN (D/6*PI);D
3030 NEXT D
3040 FOR I=1 TO 60

```

```

3050 PLOT 33+20*SIN (I/30*PI),22
+20*COS (I/30*PI)
3060 NEXT I
3070 DIM A$(704)
3080 LET X=1
3090 LET L=PEEK 16396+256*PEEK 1
6397+1
3100 FOR L=L TO L+725

```

```

3110 IF PEEK L=110 THEN GL
0
3120 LET A$(X)=CHR$ PEEK L
3130 LET X=X+1
3140 NEXT L
3150 RETURN

```

TIME BOMB LISTING

WORLD GEOGRAPHY

Shawn Byrne
TS1000 Machine Code (16K memory)
Contest entry

This program embodies a very elegant graphics routine. The program offers you two categories of study: Countries/Islands or Continents/Oceans/Seas; whichever category is chosen, you have the option of Review/Identify. If you choose the Identify option, you are asked to give the names of various geographical entities within the category you chose. If you choose Review, then you will see the names of those identities displayed as they are pointed out on the map.

It's the map that makes the difference. It is hard to believe that the block graphics of the TS1000 could give the degree of detail that Mr. Byrne has developed. And the scrolling action alone is enough to provoke wonder.

Lines 5012 and 5132 give one more option among do-nothing commands for entering the realm of USR. Since a repeated RANDOMIZE USR would destroy the randomness which is sought in lines 5030 and 5280, Mr. Byrne uses the simple IF USR...THEN (that is, if USR...<> zero, THEN), and then does nothing at all (REM), regardless of what the answer is.

The decimal values of the machine code bytes in line 1 REM (memory addresses 16514 to 16866) and in line 2 REM (16873 to 16906) are shown in the figures.

The shortcomings of the program are those inherent in any computer quiz. First, of course, the number of questions must be limited. More seriously, the computer can not tolerate any deviation from the answer it wants. If, for instance, you give *Islas Maldivas* as the answer for *Falkland Islands*, you get the hook. Similarly, you are not allowed to give *England*, the *United Kingdom*, or *Great Britain* when the computer expects "*British Isles*". Even "*The British Isles*" would be rejected. And misspellings also disqualify you--there's no such thing as a near miss.

Now there's where we really need Artificial Intelligence.

```

1 REM X?S.  ,BY?M?INKEY$MLEN
INKEY$Y?M,INKEY$MPEEK INKEY$TAN
Y?M?INKEY$MLEN INKEY$Y?M,INKEY$
MPEEK INKEY$ GOSUB ?RAND?RANDACS
?CGACS ?U?RAND4? RETURN ,COS USS
? RUN ? GOSUB ?/? RETURN COS
X$?/?/?5 GOSUB PI? RUN ? GOSUB
?/?/? ACS ?U?RAND40? RETURN ?4?/?M
RAND(?/?/?Y?/?3 GOSUB ?/?7?/?X4 PRINT
/?X RETURN COPY 4 Y?M?RAND5?/?/?
FY?/?3 GOSUB ?FF,?X4 PRINT /?M?AN
D GOSUB ?RAND,?U?RAND RETURN 64 Y?
?/?/?E?RAND? GOSUB ?/? VAL ?CHR$

```



```

,AT S?U?LN ?RAND; ( CLEAR FAST ?U?
RANDACS ?CU FAST E?RAND7ACS ?4, ) R
UN , FOR LPRINT ,4Y?/?4 LPRINT
FAST ? GOSUB ? FOR ?ACS ?C, ACS
RETURN ACS ? FOR GOSUB ?/? SLOW
AT TAN AT GOSUB ?RANDACS ?4?Y3?
;?2?/?5 GOSUB ?R FAST E?RAND7K?23
; FOR LPRINT ?/?/?Y;?4?LN ?RANDST
A$ )/? GOSUB ?SGN ? GOSUB ? FOR
?ACS ?C,ACS RETURN ACS ? FOR GOS
UB ?STR$ )? ; STOP 25;SGN FOR ?(
PEEK TAN
2 REM M$5 DIM INKEY$6-RANDLN 0
)777 FAST E?RAND)7 ;SGN ?3 GOSUB
?7:4 GOSUB ?TAN
4 REM *** WORLD GEOGRAPHY ***
BY
5 RAND
7 SLOW
10 LET DF=PEEK 16396+256*PEEK
16397
20 LET GET=DF-289
30 LET M=16540
40 LET M$=""

```

```

100 PRINT "DO YOU WANT TO ";TAB
2;"?REVIEW";TAB 2;"?IDENTIFY"
;AT 4,2;"PRESS 1 OR 2";AT 4,0;
110 LET G$=""
112 IF NOT USR GET THEN GOTO 11
2
113 REM 1,2
120 PRINT AT 4,2;" "
130 LET O$=G$

```



```

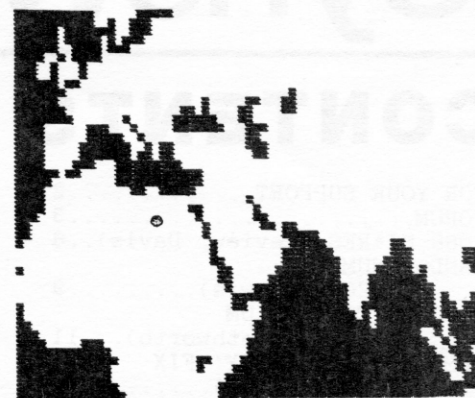
1040 NEXT A
1060 DIM P$(NN,2)
1070 LET A$=" "
;H$9"+CHR$ 11+CHR$ 11+"?28( SAV
E U$2$ " +CHR$ 69+"2"+CHR$ 72+"
+ CLEAR S$
1080 FOR A=1 TO NN
1090 LET P$(A)=A$(2*A-1 TO )
1100 NEXT A
1110 GOTO 5000
2000 LET LN=18
2002 LET NN=17
2016 DIM N$(NN,LN)
2018 LET A$="NORTH AMERICA $S
OUTH AMERICA $AFRICA
EUROPE $ASIA
AUSTRALIA $PACIFIC
OCEAN $ATLANTIC OCEAN :IND
IAN OCEAN $ARCTIC OCEAN
EMEDITERRANEAN SEA:GREAT LAKES
"+CHR$ 11+"GULF OF MEXICO
:RED SEA $HUDSON BAY
$BERING STRAIT $PERSIAN G
ULF
2020 FOR A=1 TO NN
2030 LET N$(A)=A$(1+LN*(A-1) TO
)
2040 NEXT A
2060 DIM P$(NN,2)
2070 LET A$=CHR$ 84+" " +CHR$ 11+
"2<2" +CHR$ 77+"":?H-+CHR$
72+" PLOT 288 $288 " +CHR$ 71+
CLEAR B"+CHR$ 11
2080 FOR A=1 TO NN
2090 LET P$(A)=A$(2*A-1 TO )
2100 NEXT A
5000 POKE 16520,0
5002 POKE 16521,0
5004 POKE 16522,36
5010 FOR A=1 TO 8
5012 IF USR M OR USR M OR USR M
OR USR M THEN REM
5014 NEXT A
5016 DIM T$(NN)
5018 LET CT=0
5030 LET D=INT (RND*NN)+1
5032 IF CODE T$(D) THEN GOTO 503
0
5034 LET T$(D)="1"
5040 LET L$=N$(D, TO CODE N$(D,L
N))
5060 LET Y=CODE P$(D,2)
5065 IF Y>128 THEN LET Y=Y-256
5070 POKE 16522,34+(Y<PEEK 16521
)
5080 FOR A=1 TO ABS (Y-PEEK 1652
1)
5090 IF USR M THEN REM
5100 NEXT A
5110 LET B=PEEK 16520-CODE P$(D)
5120 POKE 16522,36-3*(B>0 AND B<
42 OR B<-42)
5130 FOR A=1 TO ABS (85*(ABS B>4
2)-ABS B)
5132 IF USR M THEN REM
5134 NEXT A
5140 LET A=USR 16875
5160 PRINT AT 1,1;
5165 IF O$="1" THEN PRINT L$
5170 LET A=Y-26*(Y>26)
5180 LET B=OF+412+33*A*(Y<1 OR Y
>26)
5190 LET F=PEEK B
5200 LET C=23+128*(F>127)
5210 FOR A=1 TO 15
5220 POKE B,C
5225 FOR D=1 TO 2
5226 NEXT D
5230 POKE B,F
5240 NEXT A
5245 IF O$="1" THEN GOTO 5500
5250 LET G$=L$+"
5260 LET G$=G$( TO USR GET)
5262 REM A-Z,
5270 IF L$<>G$ THEN GOTO 5300
5280 LET A=INT (RND*3)*6+1
5282 PRINT AT 2,1;"RIGHT,GREAT.S
UPER."(A TO A+5)
5290 FOR A=1 TO 12
5292 NEXT A
5295 GOTO 5500
5300 PRINT AT 2,1;"NO, THIS IS "
;L$;"
5310 FOR A=1 TO 35
5312 NEXT A
5500 LET CT=CT+1
5502 IF CT=NN THEN GOTO 5000
5510 PRINT AT 1,1;M$
5520 GOTO 5030
9998 SAVE "GE"
9999 RUN

```

LISTING FOR GEOGRAPHY

61	91	56	4	0	1	7	26
36	62	111	50	122	65	50	108
65	62	103	50	135	65	50	211
65	201	62	103	50	122	65	50
198	65	62	111	50	135	65	50
211	65	237	91	12	64	58	138
64	203	79	40	44	203	71	58
137	64	32	15	254	26	200	60
33	33	0	25	1	247	2	237
176	24	91	254	0	200	61	33
23	3	25	84	93	1	33	0
237	65	1	247	2	237	184	24
69	6	0	203	71	58	135	64
32	28	60	254	85	32	1	175
50	136	64	19	98	107	35	62
24	14	31	237	176	35	35	19
19	61	32	245	24	35	61	254
255	32	2	62	84	50	136	64
33	23	3	25	84	93	43	62
24	14	31	237	184	43	43	27
27	61	32	245	24	3	50	137
64	237	75	136	64	4	58	138
64	254	34	32	4	62	23	128
71	17	195	9	42	12	64	175
237	82	17	85	0	197	120	214
26	193	56	5	60	71	205	139
64	25	16	253	229	9	58	138
64	203	79	40	47	229	42	12
64	35	203	71	32	4	17	247
2	25	235	225	6	32	62	86
12	185	32	2	225	229	175	237
111	235	119	203	95	40	4	203
254	203	158	235	237	103	35	19
16	228	193	201	193	237	75	136
64	203	71	32	13	62	31	95
25	129	30	85	187	56	3	237
82	55	229	42	12	64	35	48
3	30	31	25	235	225	72	6
24	24	15	62	25	185	32	10
205	139	64	213	17	77	8	237
82	209	175	237	111	235	119	203
95	40	4	203	254	203	158	235
237	103	213	17	85	0	25	227
30	33	25	209	235	12	16	211
201							

BYTES IN LINE 1 REM



50	13	33	233	65	34	22	64
205	28	17	35	35	35	229	42
12	64	17	35	0	25	209	1
31	0	237	176	35	14	32	237
176	201						

BYTES IN LINE 2 REM